IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	§ Group Art Unit: 2432
Matthew P. Duggan, *et al.*	§
	§ Examiner: Kim, Jung W.
Serial No.: 10/815,213	§
	§ Atty Docket No.: AUS920040010US1
Filed: 03/31/2004	§
	§ Customer No.: 34533
Title: Cross Domain Security	§
Information Conversion	§ Confirmation No.: 7107

Mail Stop: Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

APPEAL BRIEF

Honorable Commissioner:

This is an Appeal Brief filed pursuant to 37 CFR § 41.37 in response to the Final Office
Action of November, 25, 2009 (hereinafter the "Office Action").

REAL PARTY IN INTEREST

The real party in interest in accordance with 37 CFR § 41.37(c)(1)(i) is the patent
assignee, International Business Machines Corporation ("IBM"), a New York corporation
having a place of business at Armonk, New York 10504.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences within the meaning of 37 CFR §
41.37(c)(1)(ii).

STATUS OF CLAIMS

Status of claims in accordance with 37 CFR § 41.37(c)(1)(iii): Twenty-eight (28) claims are filed in the original application in this case. Claims 1-28 are rejected in the Office Action. Claims 1-28 are on appeal.

STATUS OF AMENDMENTS

Status of amendments in accordance with 37 CFR § 41.37(c)(1)(iv): No amendments were submitted after final rejection. The claims as currently presented are included in the Appendix of Claims that accompanies this Appeal Brief.

SUMMARY OF CLAIMED SUBJECT MATTER

Appellants provide the following concise summary of the claimed subject matter according to 37 CFR § 41.37(c)(1)(v). This summary includes a concise explanation of the subject matter defined in each of the independent claims involved in the appeal and dependent claims argued separately and includes references to the specification by page and line number and to the drawings by elements. The three independent claims involved in this appeal are claims 1, 10, and 19. Claims 1 is a method claim. Claims 10 and 19 recite counterpart aspects of the method of claim 1. Claim 10 recites system aspects of the method of claim 1. Claim 19 recites computer program product aspects of the method of claim 1. Dependent claims separately argued in this Appeal include claims 2, 7-9, 11, 16-18, 20, 25-26, and 28.

Claim 1 recites a computer-implemented method for cross domain security information conversion (page 12, lines 28-29; Figure 2). The computer includes a computer processor and a computer memory operatively coupled to the computer processor, where the computer memory has disposed within it computer program instructions that execute the method of claim 1 (page 6, lines 6-16). The method of claim 1 includes receiving from a system entity, in a security service, security information in a native format of a first security domain regarding a system entity having an identity in at least one security

2

domain, where the system entity is automated computing machinery (page 12, line 30 –
page 13, line 1; Figure 2, elements 202, 110, 102, and 212). The method of claim 1 also
includes translating the security information to a canonical format for security
information, wherein the canonical format is a data format for security information that is
standardized for use in data transformations of security information (page 15, lines 20-21;
Figure 2, elements 204 and 214). The method of claim 1 also includes transforming the
security information in the canonical format using a predefined mapping from the first
security domain to a second security domain (page 19, lines 11-13; Figure 2, elements
206 and 214). The method of claim 1 also includes translating the transformed security
information in the canonical format to a native format of the second security domain
(page 24, lines 4-6; Figure 2, elements 208 and 216). The method of claim 1 also
includes returning to the system entity the security information in the native format of the
second security domain (page 24, lines 14-16; Figure 2, elements 210 and 218).

Claim 10 recites a system for cross domain security information conversion (page 12,
lines 28-29; Figure 2). The system of claim 10 includes a computer processor operatively
coupled to a computer memory where the computer memory has disposed within it
computer program instructions for cross domain security information conversion (page 6,
lines 6-16). The system of claim 10 includes computer program instructions capable of
receiving from a system entity, in a security service, security information in a native
format of a first security domain regarding a system entity having an identity in at least
one security domain (page 12, line 30 – page 13, line 1; Figure 2, elements 202, 110, 102,
and 212). The system of claim 10 also includes computer program instructions capable
of translating the security information to a canonical format for security information
(page 15, lines 20-21; Figure 2, elements 204 and 214). The system of claim 10 also
includes computer program instructions capable of transforming the security information
in the canonical format using a predefined mapping from the first security domain to a
second security domain (page 19, lines 11-13; Figure 2, elements 206 and 214). The
system of claim 10 also includes computer program instructions capable of translating the
transformed security information in the canonical format to a native format of the second
security domain (page 24, lines 4-6; Figure 2, elements 208 and 216). The system of

claim 10 also includes computer program instructions capable of returning to the system entity the security information in the native format of the second security domain (page 24, lines 14-16; Figure 2, elements 210 and 218).

Claim 19 recites a computer program product for cross domain security information conversion (page 12, lines 28-29; Figure 2). The computer program product of claim 19 is embodied on a recordable computer-readable medium (page 6, lines 18-19). The computer program product of claim 19 includes computer program instructions which when installed and executed on a data processing system are capable of causing the data processing system to carry out the step of receiving from a computer program product entity, in a security service, security information in a native format of a first security domain regarding a computer program product entity having an identity in at least one security domain, wherein the system entity is automated computing machinery (page 12, line 30 – page 13, line 1; Figure 2, elements 202, 110, 102, and 212). The computer program product of claim 19 also includes computer program instructions that carry out the step of translating the security information to a canonical format for security information (page 15, lines 20-21; Figure 2, elements 204 and 214). The computer program product of claim 19 also includes computer program instructions that carry out the step of transforming the security information in the canonical format using a predefined mapping from the first security domain to a second security domain (page 19, lines 11-13; Figure 2, elements 206 and 214). The computer program product of claim 19 also includes computer program instructions that carry out the step of translating the transformed security information in the canonical format to a native format of the second security domain (page 24, lines 4-6; Figure 2, elements 208 and 216). The computer program product of claim 19 also includes computer program instructions that carry out the step of returning to the computer program product entity the security information in the native format of the second security domain (page 24, lines 14-16; Figure 2, elements 210 and 218).

Claim 2 recites a method where transforming the security information includes structure transformation and value transformation and mapping a system entity's identity in the

first security domain to a another identity in the second security domain (page 19, lines 11-23 and Figure 2, elements 206, 213, and 205).

Claim 7 recites a method were the native format of the first security domain is expressed in XML, the canonical format is expressed in XML, and translating the security information in a native format of a first security domain to a canonical format is carried out in dependence upon a mapping, expressed in XSL, from the native format of the first security domain to a canonical format (page 17, lines 23-31 and Figure 2, elements 204, and 214).

Claim 8 recites a method where the canonical format is expressed in XML and the predefined mapping from the first security domain to a second security domain is expressed in XSL (page 19, lines 11-23 and Figure 2, elements 206, 213, and 205).

Claim 9 recites a method where the second native format is expressed in XML, the canonical format is expressed in XML, and translating the transformed security information in the canonical format to a native format of the second security domain is carried out in dependence upon a predefined mapping, expressed in XSL, from the canonical format to the native format of the second security domain (page 24, lines 4-16 and Figure 2, elements 208, 216, 210 and 218).

Claim 11 recites a system where transforming the security information includes structure transformation and value transformation, including mapping a system entity's identity in the first security domain to a another identity in the second security domain (page 19, lines 11-23 and Figure 2, elements 206, 213, and 205).

Claim 16 recites a system where translating the security information in a native format of a first security domain to a canonical format comprises a mapping, expressed in XSL, from the native format of the first security domain to a canonical format (page 17, lines 23-31 and Figure 2, elements 204, and 214).

5

Claim 17 recites a system where the canonical format is expressed in XML and the predefined mapping from the first security domain to a second security domain is expressed in XSL(page 19, lines 11-23 and Figure 2, elements 206, 213, and 205).

Claim 18 recites a system where the second native format is expressed in XML, the canonical format is expressed in XML, and translating the transformed security information in the canonical format to a native format of the second security domain comprises a predefined mapping, expressed in XSL, from the canonical format to the native format of the second security domain (page 24, lines 4-16 and Figure 2, elements 208, 216, 210 and 218).

Claim 20 recites a computer program product where computer program instructions for transforming the security information include computer program instructions for structure transformation and value transformation, including computer program instructions for mapping a system entity's identity in the first security domain to another identity in the second security domain (page 19, lines 11-23 and Figure 2, elements 206, 213, and 205).

Claim 25 recites a computer program product where computer program instructions for translating the security information in a native format of a first security domain to a canonical format comprises a mapping, expressed in XSL, from the native format of the first security domain to a canonical format (page 17, lines 23-31 and Figure 2, elements 204, and 214).

Claim 26 recites a computer program product where the canonical format is expressed in XML and the predefined mapping from the first security domain to a second security domain is expressed in XSL(page 19, lines 11-23 and Figure 2, elements 206, 213, and 205).

Claim 28 recites a computer program product where the second native format is expressed in XML, the canonical format is expressed in XML, and computer program instructions for translating the transformed security information in the canonical format

to a native format of the second security domain comprises a predefined mapping, expressed in XSL, from the canonical format to the native format of the second security domain (page 24, lines 4-16 and Figure 2, elements 208, 216, 210 and 218).

GROUNDS OF REJECTION

In accordance with 37 CFR § 41.37(c)(1)(vi), Appellants provide the following concise statement for each ground of rejection:

1. Claims 1-28 stand rejected under 35 U.S.C. § 103 as being unpatentable under Dunn, et al. (U.S. Pat. No. 7,428,750) in view of Bussler, et al. (U.S. Pat. No. 7,072,898).

ARGUMENT

Appellants present the following argument pursuant to 37 CFR § 41.37(c)(1)(vii) regarding the ground of rejection on appeal in the present case.

Argument Regarding The First Ground Of Rejection On Appeal: Claims 1-28 Stand Rejected Under 35 U.S.C. § 103 As Being Unpatentable Under Dunn In View Of Bussler

Claims 1-28 again stand rejected under 35 U.S.C. § 103(a) as unpatentable over Dunn in view of Bussler. To establish a prima facie case of obviousness, the proposed combination of Dunn and Bussler must teach or suggest all of Appellants' claim limitations. MPEP 2142 (citing *In re Royka*, 490 F.2d 981, 985, 180 USPQ 580, 583 (CCPA 1974)). As shown below in more detail, the proposed combination of Dunn and Bussler cannot establish a prima facie case of obviousness because the proposed combination does not teach or suggest each and every element of the claims of the present application. As such, Appellants respectfully traverse each rejection individually.

Claim 1 recites a computer-implemented method for cross domain security information conversion, that includes among other elements, receiving from a system entity, in a

security service, security information in a native format of a first security domain, transforming the security information in a canonical format using a predefined mapping from the first security domain to a second security domain; and returning to the system entity the security information in the native format of the second security domain. That is, claim 1 recites security information conversion from security information in the native format of the first second security domain into a native format required by a second domain. To accomplish this security information conversion, the claims of the present application recite transforming the security information in a canonical format using a predefined mapping from the first security domain to a second security domain. Appellants respectfully submit that neither Dunn nor Bussler, either alone or in combination, discloses a predefined mapping from a first security domain to a second security domain to carry out the claimed transformation of security information.

The Office Action admits at page 8 that Dunn does not disclose "transforming the security information in the canonical format to a native format." Because such a transformation is carried out through use of a predefined mapping from a first security domain to a second security domain it is no surprise that Dunn does not disclose such a predefined mapping as claimed here. Instead, Dunn discloses associations among separate user identities and authentication tickets of a particular user. These associations, however, are not used in a transformation, as admitted by the Office Action, but are instead used as an index for a lookup. Dunn describes the use of such identity mappings at column 19, lines 17-47, reproduced here for convenience of reference:

> 1. User A registers his pageA.net and pageB.net identities with the identity broker 206.

> 2. User A signs in to the pageB.net mail server using the pageA.net authentication ticket.

> 3. The pageB.net mail server sends a request to an authentication system to confirm the pageA.net identity.

> 4. The pageB.net mail server finds that it cannot use the pageA.net authentication ticket.

5. The pageB.net mail server sends the pageA.net authentication ticket to the identity broker 206 specified in the authentication ticket.

6. The identity broker 206 reviews the registered identities associated with the pageA.net identity and finds that the pageB.net identity may be used for mail.

7. The identity broker 206 requests the authentication system to authenticate the pageB.net identity.

8. The identity broker 206 returns the pageB.net identity to the pageB.net mail server in an authenticated manner.

9. The pageB.net mail server packages User A's mail messages for delivery to User A.

10. The pageB.net mail server requests the identity broker 206 to reset the authentication ticket for the pageA.net service.

11. The identity broker 206 fetches the appropriate authentication ticket for use in returning the packaged mail messages through the pageA.net firewall.

12. User A's pageA.net server accepts the inbound message because it contains an authenticated pageA.net identity.

13. User A reviews the requested email from the pageB.net mail server in the accepted inbound message.

Emphasis added. In the response to arguments section on page 4 of the Office Action, the Office Action takes the position that Dunn at column 19, lines 17-35 discloses transforming the security information including "a value transformation for mapping a system's entity's identity in the first security domain to another identity in the second security domain (steps5-6)." Appellants initially submit that the Office Action presents contrary positions: At page 8, in the primary remarks regarding the rejection, the Office Action takes the position Dunn does not teach the claimed transformation while at page 4, in the response to arguments section, the Office Action takes the position that Dunn does teach the claimed transformation. Appellants further submit that, as can be seen from Dunn's step 6 above, the associations among identities and authentication tickets of webpages are used to identify, for a particular user, an identity and authentication ticket

for a different webpage. Identifying an identity and authentication ticket does not disclose transforming anything. Dunn's associations are not used to transform security data as claimed here. As such, Dunn does not disclose or suggest a predefined mapping from a first security domain to a second security domain to carry out the claimed transformation of security information.

Bussler also does not disclose or suggest a predefined mapping from a first security domain to a second security domain as claimed here. At column 4, lines 3-11, Bussler discloses an exchange from a source application to a target application, a transformation of data from a source application format to a target application format:

> According to one embodiment, each item exchanged from a source application to a target application goes through five distinct processing phases, referred to herein as:
> 1) source-side native phase
> 2) source-side application phase
> 3) common view phase
> 4) target-side application phase
> 5) target-side native phase

Bussler's exchange, however, is not carried out with a predefined mapping from a first security domain to a second security domain as claimed here. In fact, Bussler is completely unconcerned with security domains and, as such, one would not except to find a mapping from one security domain to another. Bussler, having not disclosed security domains, cannot possibly disclose a predefined mapping from a first security domain to a second security domain as claimed here.

Because neither Dunn nor Bussler, either alone or in combination, discloses or suggests a predefined mapping from a first security domain to a second security domain, the proposed combination of Dunn and Bussler cannot support a prima facie case of obviousness against claim 1 of the present application.

Relations Among Claims

Independent claims 10 and 19 recite respectively system and computer program product aspects of the method of claim 1. As explained above in detail, the combination of Dunn and Bussler does not render claim 1 obvious. For the same reasons that the combination of Dunn and Bussler does not render obvious claim 1, the combination of Dunn and Bussler also does not render obvious a system or a computer program product for cross domain security information conversion corresponding to independent claims 10 and 19. Independent claims 10 and 19 are therefore patentable and should be allowed.

Dependent Claims 2-9, 11-18, and 20-28 depend from independent claims 1, 10, and 19. Each dependent claim includes all of the limitations of the independent claim from which it depends. Because the combination of Dunn and Bussler does not render obvious the independent claims, the combination of Dunn and Bussler also does not render obvious the dependent claims of the present application, which are further limitations of the independent claims. As such, claims 2-9, 11-18, and 20-28 are also patentable and should be allowed at least by virtue of their dependence on allowable claims.

The Proposed Combination Of Dunn And Bussler Does Not Disclose Or Suggest Each And Every Element And Limitation Of The Dependent Claims

In addition to the fact that the proposed combination of Dunn and Bussler does not render the independent claims obvious, the proposed combination of Dunn and Bussler also fails to render many of the dependent claims obvious. Claims 2, 11, and 20 which recite, among other elements, a "structure transformation and value transformation" when transforming the security information. Appellants submit that neither Dunn nor Bussler discloses a structure transformation and value transformation carried out with respect to security information as claimed here. Instead, as explained above, Dunn discloses identifying another identity for a particular user and Bussler discloses a transformation among application formats. Although Dunn's identification of another identity for a particular user may arguably be viewed as a value transformation, Dunn makes no structure transformation whatsoever. Bussler's transformation does not disclose a

structure transformation as claimed here because *as claimed* the structure transformation is part of a transformation of security information and Bussler's transformation a transformation among application formats – not security information. Because neither Bussler nor Dunn, either alone or in combination, disclose or suggest a structure transformation when transforming the security information, the proposed combination cannot support a prima facie case of obviousness against claims 2, 11, and 20.

Each of claims 7-9, 16-18, 25-26, and 28 recites XSL. XSL is described in Appellants' original specification at page 8, lines 14-20:

> "XSL" refers to the 'Extensible Style Language,' family of recommendations for defining XML document transformation and presentation, including a language for transforming XML. XSL supports specifications or mappings allowing users or developers to transform XML documents across different applications. XSL provides the capability of specifying transformations of data and data structures, including, for example, canonical formats as well as some native security information formats, expressed in XML.

Neither Dunn nor Bussler disclose or suggest such an Extensible Style Language that defines XML document transformation. The Office Action at page 10 takes Official Notice that XML, not XSL, is a well known security information format. The Office Action does not even mention XSL with respect to the Official Notice. In an attempt to preempt the use of Official Notice with respect to XSL, Appellants submit that the Office Action should not take Official Notice of the use of XSL as claimed in the present application without documentary evidence because the use of XSL as claimed is not capable of "instant and unquestionable demonstration as being well-known." See MPEP 2144.03(A). The fact that the claimed use of XSL is not capable of instant and unquestionable demonstration as being well-known is evidenced by the fact that the two references relied on by the Office Action in rejecting the claims – ostensibly the references most similar in subject matter – fail completely to recite XML, XSL, or any variation thereof. Having not disclosed XSL the proposed combination of Dunn and Bussler cannot disclose any of claims 7-9, 16-18, 25-26, and 28 and Official Notice with respect to XSL is lacking without documentary evidence to support such notice. Further,

even if XSL is generally well known, Appellants submit that there is no teaching or suggestion in either Dunn or Bussler, nor any reasoning in the Office Action, that XSL is suitable for performing a transformation as claimed in the present application. As such, Appellants therefore request the rejections of claims 7-9, 16-18, 25-26, and 28 be withdrawn.

In view of the arguments above, reversal on the ground of rejection is requested.

The Commissioner is hereby authorized to charge or credit Deposit Account No. 09-0447 for any fees required or overpaid.

Respectfully submitted,

Date: March 5, 2010   By:

Brandon C. Kennedy
Reg. No. 61,471
Biggers & Ohanian, LLP
P.O. Box 1469
Austin, Texas 78767-1469
Tel. (512) 472-9881
Fax (512) 472-9887
ATTORNEY FOR APPELLANTS

**APPENDIX OF CLAIMS
ON APPEAL IN PATENT APPLICATION OF
MATTHEW PAUL DUGGAN, *ET AL.*, SERIAL NO. 10/815,213**

<u>CLAIMS</u>

What is claimed is:

1. (Previously Presented) A computer-implemented method for cross domain
    security information conversion, the computer comprising a computer processor
    and a computer memory operatively coupled to the computer processor, the
    computer memory having disposed within it computer program instructions that
    execute the method, the method comprising:

    receiving from a system entity, in a security service, security information in a
    native format of a first security domain regarding a system entity having an
    identity in at least one security domain, wherein the system entity comprises
    automated computing machinery;

    translating the security information to a canonical format for security information,
    wherein the canonical format is a data format for security information that is
    standardized for use in data transformations of security information;

    transforming the security information in the canonical format using a predefined
    mapping from the first security domain to a second security domain;

    translating the transformed security information in the canonical format to a
    native format of the second security domain; and

    returning to the system entity the security information in the native format of the
    second security domain.

2. (Original) The method of claim 1 wherein transforming the security information includes structure transformation and value transformation, including mapping a system entity's identity in the first security domain to a another identity in the second security domain.

3. (Original) The method of claim 1 wherein receiving security information further comprises receiving a request for security information for the second security domain, wherein the request encapsulates the security information in a native format of a first security domain.

4. (Original) The method of claim 3 wherein the system entity comprises a system entity requesting access to a resource in the second security domain.

5. (Original) The method of claim 3 wherein the system entity comprises a system entity providing access to a resource in the second security domain.

6. (Original) The method of claim 1 wherein translating the security information in a native format of a first security domain to a canonical format is carried out through a procedural software function.

7. (Original) The method of claim 1 wherein the native format of the first security domain is expressed in XML, the canonical format is expressed in XML, and translating the security information in a native format of a first security domain to a canonical format is carried out in dependence upon a mapping, expressed in XSL, from the native format of the first security domain to a canonical format.

8. (Original) The method of claim 1 wherein the canonical format is expressed in XML and the predefined mapping from the first security domain to a second security domain is expressed in XSL.

9.   (Original) The method of claim 1 wherein the second native format is expressed in XML, the canonical format is expressed in XML, and translating the transformed security information in the canonical format to a native format of the second security domain is carried out in dependence upon a predefined mapping, expressed in XSL, from the canonical format to the native format of the second security domain.

10.   (Previously Presented) A system for cross domain security information conversion, the system comprising a computer processor operatively coupled to a computer memory, the computer memory having disposed within it computer program instructions for:

        receiving from a system entity, in a security service, security information in a native format of a first security domain regarding a system entity having an identity in at least one security domain;

        translating the security information to a canonical format for security information;

        transforming the security information in the canonical format using a predefined mapping from the first security domain to a second security domain;

        translating the transformed security information in the canonical format to a native format of the second security domain; and

        returning to the system entity the security information in the native format of the second security domain.

11.   (Previously Presented) The system of claim 10 wherein transforming the security information includes structure transformation and value transformation, including mapping a system entity's identity in the first security domain to a another identity in the second security domain.

16

12. (Previously Presented) The system of claim 10 wherein receiving security information further comprises receiving a request for security information for the second security domain, wherein the request encapsulates the security information in a native format of a first security domain.

13. (Original) The system of claim 12 wherein the system entity comprises a system entity requesting access to a resource in the second security domain.

14. (Original) The system of claim 12 wherein the system entity comprises a system entity providing access to a resource in the second security domain.

15. (Previously Presented) The system of claim 10 wherein translating the security information in a native format of a first security domain to a canonical format comprises a procedural software function.

16. (Previously Presented) The system of claim 10 wherein translating the security information in a native format of a first security domain to a canonical format comprises a mapping, expressed in XSL, from the native format of the first security domain to a canonical format.

17. (Original) The system of claim 10 wherein the canonical format is expressed in XML and the predefined mapping from the first security domain to a second security domain is expressed in XSL.

18. (Previously Presented) The system of claim 10 wherein the second native format is expressed in XML, the canonical format is expressed in XML, and translating the transformed security information in the canonical format to a native format of the second security domain comprises a predefined mapping, expressed in XSL, from the canonical format to the native format of the second security domain.

19.   (Previously Presented) A computer program product for cross domain security information conversion, the computer program product embodied on a recordable computer-readable medium, the computer program product comprising computer program instructions which when installed and executed on a data processing system, are capable causing the data processing system to carry out the steps of:

receiving from system entity, in a security service, security information in a native format of a first security domain regarding a system entity having an identity in at least one security domain, wherein the system entity comprises automated computing machinery

translating the security information to a canonical format for security information;

transforming the security information in the canonical format using a predefined mapping from the first security domain to a second security domain;

translating the transformed security information in the canonical format to a native format of the second security domain; and

returning to the system entity the security information in the native format of the second security domain.

20.   (Previously Presented) The computer program product of claim 19 wherein computer program instructions for transforming the security information includes computer program instructions for structure transformation and value transformation, including computer program instructions for mapping a system entity's identity in the first security domain to another identity in the second security domain.

21.   (Previously Presented) The computer program product of claim 19 wherein computer program instructions for receiving security information further

comprises computer program instructions for receiving a request for security information for the second security domain, wherein the request encapsulates the security information in a native format of a first security domain.

22.   (Previously Presented) The computer program product of claim 21 wherein the system entity comprises a system entity requesting access to a resource in the second security domain.

23.   (Previously Presented) The computer program product of claim 21 wherein the system entity comprises a system entity providing access to a resource in the second security domain.

24.   (Previously Presented) The computer program product of claim 19 wherein computer program instructions for translating the security information in a native format of a first security domain to a canonical format comprises a procedural software function.

25.   (Previously Presented) The computer program product of claim 19 wherein computer program instructions for translating the security information in a native format of a first security domain to a canonical format comprises a mapping, expressed in XSL, from the native format of the first security domain to a canonical format.

26.   (Original) The computer program product of claim 19 wherein the canonical format is expressed in XML and the predefined mapping from the first security domain to a second security domain is expressed in XSL.

27.   (Previously Presented) The computer program product of claim 19 wherein computer program instructions for translating the transformed security information in the canonical format to a native format of the second security domain comprises a procedural software function.

28. (Previously Presented) The computer program product of claim 19 wherein the second native format is expressed in XML, the canonical format is expressed in XML, and computer program instructions for translating the transformed security information in the canonical format to a native format of the second security domain comprises a predefined mapping, expressed in XSL, from the canonical format to the native format of the second security domain.

.

**APPENDIX OF EVIDENCE
ON APPEAL IN PATENT APPLICATION OF
MATTHEW PAUL DUGGAN, *ET AL.*, SERIAL NO. 10,815,213**

This is an evidence appendix in accordance with 37 CFR § 41.37(c)(1)(ix).

There is in this case no evidence submitted pursuant to 37 CFR §§ 1.130, 1.131, or 1.132, nor is there in this case any other evidence entered by the examiner and relied upon by the Appellants.

RELATED PROCEEDINGS APPENDIX

This is a related proceedings appendix in accordance with 37 CFR § 41.37(c)(1)(x).
There are no decisions rendered by a court or the Board in any proceeding identified
pursuant to 37 CFR § 41.37(c)(1)(ii).